

# Statistical Arbitrage Trading with Wavelets and Artificial Neural Networks

Christopher Zapart

Advanced Financial Trading Solutions Ltd., 9 Dundas Mews,  
Middlesex, Enfield EN3 6YA, United Kingdom  
Email: chris.zapart@afts-online.co.uk

**Abstract:** *The paper outlines the use of an alternative option pricing scheme to perform statistical arbitrage in derivative markets. The method links a binomial tree to an innovative stochastic volatility model that is based on wavelets and artificial neural networks. Wavelets provide a convenient signal/noise decomposition of volatility in a non-linear feature space. Neural networks are used to infer future volatility levels from the wavelets feature space in an iterative manner. The bootstrap method provides 95% confidence intervals for the option prices. When used to set up delta-hedged arbitrage trades in the US equity options market, the proposed approach generates substantial profits.*

**Keywords:** *Artificial neural networks, binomial trees, Black-Scholes formulæ, bootstrap, delta-hedging, option pricing, statistical arbitrage, stochastic volatility, wavelets*

## 1 INTRODUCTION

The Black-Scholes formulæ [1], though widely accepted, has been derived under several strict assumptions, mainly: log-normal distribution of share price returns, stock prices following a random walk, constant volatility and a presence of a rational investor. Its estimates are valid only for a short period of time, which leads to a need for dynamic delta hedging with its increased trading costs. These underlying assumptions are quite restrictive. Indeed, they have been questioned by researchers and market practitioners on a number of occasions. The evidence can be provided by the emergence of alternative models such as stochastic volatility (e.g. GARCH) [7] or constant elasticity of variance model [2]. In all studies, a measure of volatility is the most important input to all pricing models, in particular binomial trees, in which it is being used in perhaps the most visible way of all the models. As such, it is imperative that correct volatility values are being used and that the assumptions upon which the models have been based are correct. However, in what is quite unsettling, measuring volatility is in itself more of an art than science. There is no clear agreement as to what measure of volatility should be used as an input to the options pricing formulæ. A simple rule of thumb [6] is to take a standard deviation of stock price returns measured over a sliding window the length of the number of days left to the expiry. For example, if an option has only five days left before it expires, it may be desirable to use volatility measured over the past five days rather than six months. The stochastic volatility models attempt to remove one major shortcoming of traditional options pricing:

the assumption of constant volatility. Even if "correct" values for the volatility are found, they are only valid for a relatively short period of time. Volatility does change along with the internal dynamics of the markets.

The purpose of this paper is to present the results of applying an alternative option pricing scheme to perform statistical arbitrage trading in the US derivatives market. The new method, described in [9], turns the stock volatility into a stochastic variable that can, to some extent, be predicted with techniques borrowed from computational artificial intelligence. The proposed solution elegantly integrates with binomial trees, which themselves are convenient for performing numerical options pricing as opposed to using closed-form analytical solutions. Wavelets have been used to solve various financial engineering problems, for example pricing of financial derivatives [3] or iterative long-term forecasting of financial time-series [5]. Their main advantage is the ability to model jumps and discontinuities present in the financial time series. Performing a signal/noise decomposition of the underlying time-series becomes a straightforward operation in the non-linear wavelets domain. In this paper wavelets are used to decompose the measure of volatility into a number of components at different time scales. Inferring future volatility is reduced to a task of iterating forward in time these components in the wavelets feature space followed by a reconstruction of the volatility back into the time domain. Artificial neural networks have been proven to be universal approximators [4]. They are easy to implement in software, perform their computations very efficiently and are simple to train. For these reasons they have therefore been chosen to model the (non-linear) generator used to iterate forward in time the individual wavelets components.

For any forecast to be meaningful ideally one should produce not only the point forecast but also estimates for the 95% confidence intervals. The only problem is that a large amount of data points would be required for a reliable estimation of the confidence intervals. In the US stock market there is typically several thousand stocks on which options are available. If the proposed new algorithm was to be used in real-time, the computational costs involved would have to be kept down to an absolute minimum. As a consequence, only a handful of forecasts could be produced for each option. Fortunately, there exists a simple technique - a bootstrap method [11] - that is able to produce reliable estimates of 95% confidence intervals from a small number of observations. In the paper we hence take full advantage of bootstrapping to produce confidence intervals for the op-

VISIT...

LANZAROTE  
*Caliente*.COM

tions prices estimated by the new model.

## 2 MODEL DESIGN

### 2.1 Binomial Trees

The binomial tree-based options pricing schemes [6] are well-researched. However, in this paper we take a fresh look at them and show how to improve their working with the help of wavelets and artificial neural networks.

Let us consider a basic binomial model (see Figure 1). Suppose that at a time step  $t$  (denoted by the point A) the stock XYZ is trading at  $x(t)$ . In the next time step  $t+1$  the stock can move from the point A to B either up or down by the absolute return  $|r(t+1)|$ . Depending on the absolute size of the move, the values of the call option at the upper and lower levels can be determined to be  $f_u$  and  $f_d$  respectively. The option price  $f$  at the point A is then given by (1) and (2) [6]:

$$f = \exp(-r_m T) [p f_u + (1-p) f_d], \quad (1)$$

$$p = \frac{\exp(r_m T) - d}{u - d}, \quad (2)$$

where  $r_m$  is the market risk free rate,  $T$  is the time to the options expiry,  $f_u$  is the upper value of the option and  $f_d$  is the lower value of the option at the point B. Equations (3) and (4) define the values  $u$  and  $d$ :

$$u = \frac{x(t) + |r(t+1)|}{x(t)}, \quad (3)$$

$$d = \frac{x(t) - |r(t+1)|}{x(t)}. \quad (4)$$

The same approach can be repeated at the point B and at all successive time steps  $t+k$ , where  $k = 1..N$  and  $N$  is the number of discrete time steps to the options expiry. Traditionally the binomial tree models would assume the daily stock price returns that give rise to the absolute returns  $|r(t+k)|$  to be a constant function of time. That is because one of the basic assumptions of the Black-Scholes model is constant volatility. The various stochastic volatility models break that assumption and make the volatility a variable function of time.

The difference between our proposed approach and the mainstream stochastic volatility models is that we use wavelets and neural networks to infer the future levels of volatility, which in addition is being measured in a non-standard way. We do not attempt to measure and forecast the volatility by taking a standard deviation of stock price returns measured over a sliding window with a variable length, which may be introducing an unwanted systematic bias. Instead we attempt to model the underlying absolute price trajectory of the stock XYZ, as expressed explicitly by the parameter  $|r(t+k)|$ , over the lifespan of a particular option. At any given time step  $t+k$  we assume the absolute return to be

$$|r(t+k)| = |x(t+k) - x(t+k-1)| \quad (5)$$

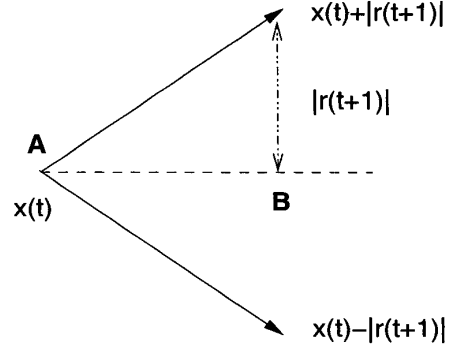


Figure 1: A simple binomial tree used to price options of the stock XYZ.  $x(t)$  is the current stock price at the point A whereas  $|r(t+1)|$  denotes the absolute price change at the next time step  $t+1$ . Point B shows two potential future outcomes: a move up and a move down in the share price.

and we attempt to produce forecasts of the relative absolute returns  $r^*(t+k)$  as defined in (6) for all the time steps leading to the options expiry:

$$r^*(t+k) = \left| \frac{x(t+k)}{x(t+k-1)} - 1 \right|, \quad (6)$$

where  $k = 1..N$ .

The parameters  $u$  and  $d$  which are needed by (2) are then assumed to be

$$u = 1 + r^*(t+k), \quad (7)$$

$$d = 1 - r^*(t+k). \quad (8)$$

The next subsection will provide an explanation of how the relative absolute returns  $r^*(t)$  are modelled with wavelets and artificial neural networks.

### 2.2 Iterative Forecasting

A detailed description of iterative forecasting of time-series with wavelets can be found in the paper by Hazarika and Lowe [5]. A similar iterative forecasting scheme with neural networks using the Principal Components Analysis (PCA) feature space has been described in [10]. For the purpose of this paper we have chosen to follow the approach based on wavelets. Initial experiments indicate that the wavelets-based feature space is better suited for analysing the stock price returns series, which is characterised by sudden jumps and discontinuities. There are three major steps involved in the wavelets-based forecasting process:

- decomposition of the time-series in the wavelets domain
- analysis and forecasting of the wavelets coefficients in the non-linear feature space (the wavelets domain)
- reconstruction of the time-series from the predicted wavelets coefficients back into the time domain.

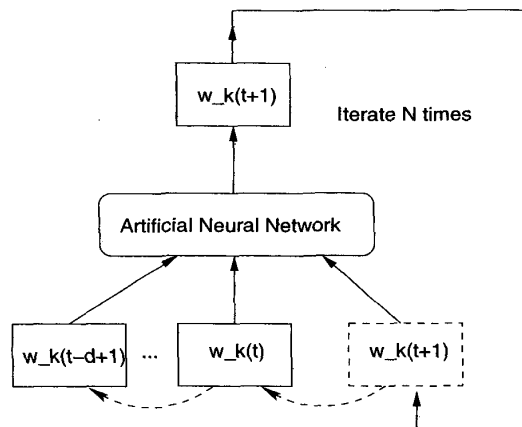


Figure 2: Forecasting the  $k$ th wavelets coefficient  $w_k(t)$   $N$  steps ahead with an artificial neural network (Variant A). The parameter  $d$  denotes the length of a delay vector containing the previous wavelets coefficients. The current output is used as an input in the successive iterations after discarding the farthest input to the left and shifting all the inputs to the left by one time step. The process is repeated  $N$  times.

The Haar wavelets are used throughout the experiments. Using different wavelet types had no noticeable effect on the performance of the models. We use a sliding window of the length of thirty two days to construct a delay matrix from the underlying signal  $r^*(t)$ . Next, a wavelet transform is applied to successive delay vectors yielding thirty two time-series containing the corresponding wavelets coefficients, of which only the first eight are retained for further processing. Each time-series with the wavelet coefficient is independently forecast by training a separate multilayered perceptron (MLP) neural network to produce a one-step ahead forecast of the particular wavelet coefficient. After training each network is fast-forwarded in time  $N$  steps by feeding its own one-step ahead forecasts back to the input layer. The Figure 2 schematically represents the forecasting process.

In the last step the first eight predicted high-order wavelets coefficients are complemented by adding twenty four zero-filled "dummy" low-order coefficients so that we get new vectors with the original number of thirty two wavelet components. An inverse wavelet transform is carried out on the zero-padded wavelet vectors in order to reconstruct the predicted underlying signal  $r^*(t)$ .

Effectively we perform an arbitrary signal/noise decomposition in the wavelets domain by retaining the first eight wavelets coefficients – regarded as the signal – and discarding the last twenty four least-significant wavelets – thought to represent the noise. There remains one issue of special importance that has not been discussed yet: choosing the right signal/noise cut-off point. Unfortunately there does not exist a reliable and exact method of choosing the right length of the delay window and the number of wavelets coefficients that should be retained for further processing. One could examine changes in the plot of eigenvalues of the SVD decomposition of the delay matrix (as in [5]) when deciding

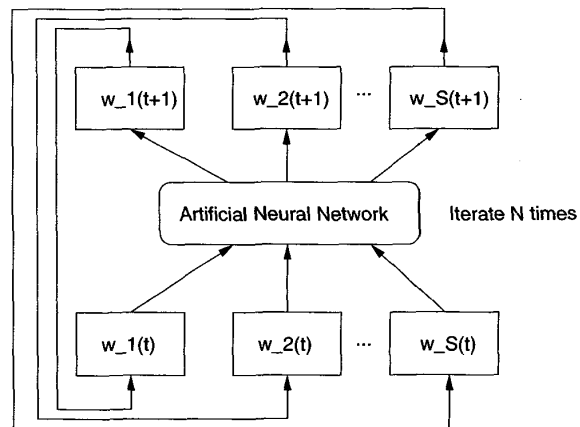


Figure 3: Forecasting the first  $S$  wavelets coefficients  $w_k(t)$   $N$  steps ahead with an artificial neural network (Variant B).

on the correct minimum length of the delay vector. It is not, however, an exact method and in many cases the parameters have to be chosen using some form of prior knowledge and by a trial and error, which has been the case in this paper.

The entire process is repeated five times for each option under consideration. Bootstrapping, when applied to the resulting small sample of options prices, produces reliable estimates for the 95% confidence intervals. Whilst ideally one would like to be using the bootstrap with more than five data points, there are physical limits on the amount of time available for repeating the calculations a large number of times.

There exists a number of alternative ways of producing forecasts of the underlying time-series based on the wavelets transform. For example, instead of using eight separate neural networks to forecast the wavelet coefficients time-series, one might want to train a single neural network using a current snapshot of all eight coefficients  $w_1(t), w_2(t), \dots, w_8(t)$  as the inputs. The network could then be trained to produce the next time step estimates of the entire vector  $w_1(t+1), w_2(t+1), \dots, w_8(t+1)$ , which would be fed back to the input layer in the successive steps (see Figure 3).

There are advantages and disadvantages to both methods. The first approach (Variant A: having a separate model for each wavelet coefficient) is computationally more intensive but it may yield more stable results. On the other hand, using only a single network (Variant B) may be more desirable when the speed of getting a solution is more important than its stability or accuracy. It may also be the case that the latter, less stable approach may actually yield improved options prices when used as part of the committees of experts approach [4], [8]. An arbitrage trading system could potentially benefit from using less stable models as they could be more likely to detect more extreme cases of options mispricing (indicating good hedging opportunities).

In the next section we present the results of applying the new technique to price and trade call options on the selected stocks traded in the US stock market. The paper presents results obtained for the two variants of the wavelets model.

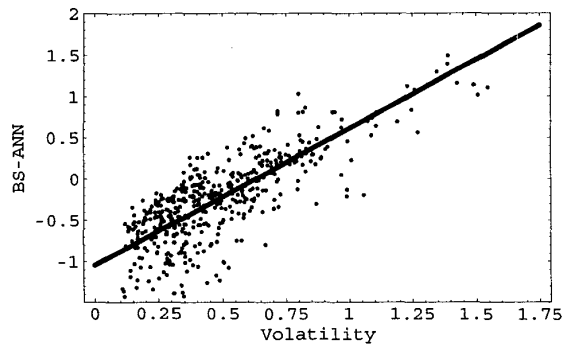


Figure 4: The difference between simulated Black-Scholes options prices ( $BS$ ) and neural network outputs ( $ANN$ ) plotted versus the historical volatility for shares of AOL Time Warner (Variant A). The dots denote data points obtained from the experiment and a solid line shows a corresponding linear regression fit.

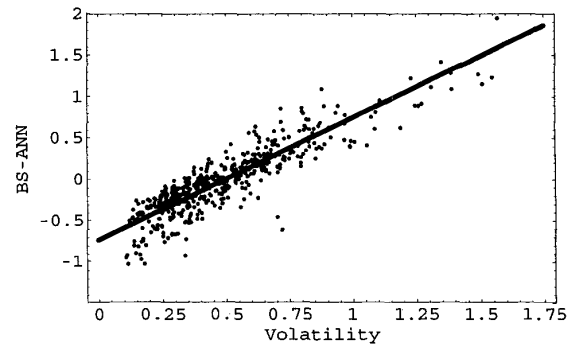


Figure 5: The difference between simulated Black-Scholes options prices ( $BS$ ) and neural network outputs ( $ANN$ ) plotted versus the historical volatility for shares of AOL Time Warner (Variant B). The dots denote data points obtained from the experiment and a solid line shows a corresponding linear regression fit.

For comparison real options prices from the Chicago Board of Exchange will be used as a proxy to the "standard" Black-Scholes model as they are most likely to reflect any improvements introduced over time to the original formulæ.

### 3 EXPERIMENTAL

The simulation results presented in the paper have been obtained using custom-built software developed by the author in C/C++ under Linux. The software simulates artificial neural networks, facilitates their training and links them to an external wavelets library. The adoption of the open-source Linux operating system has made it very convenient and cost-effective to operate a parallel Beowulf high-performance computing cluster, which is used to speed up training of neural networks. Mathematica 4.1 for Linux from Wolfram Research has also proved to be an invaluable research tool. In addition, a standalone Java code automatically scans the Internet and downloads the latest options prices directly from the US options exchanges, subsequently storing them in a local open-source MySQL database.

#### 3.1 The Proof of Concept

In order to establish the validity of the approach, initial experiments have been carried out using five hundred simulated Black-Scholes call option prices for companies traded on the US stock market. On each trading day between the days 4/14/2000 and 4/17/2002 one delta-hedged call option is written with a strike price set exactly at the money<sup>1</sup> and five trading days left until a simulated expiry day. A historical volatility as measured over the previous five days is fed into a vanilla Black-Scholes formulæ. Such simulated options prices are compared with outputs from artificial neural networks using two variants of the wavelet predictors as has been shown in Figures 2 and 3.

<sup>1</sup>Options with the strike price very close to the current value of the underlying instrument (company share price)

The sample results obtained for the shares of AOL Time Warner are presented in Figures 4 and 5, which plot the difference between simulated Black-Scholes prices  $BS$  and artificial neural networks estimates  $ANN$  as a function of the historical volatility. A very clear pattern emerges: the higher the volatility the bigger the difference between Black-Scholes and neural network options prices, which leads us to speculate that

*Assuming the data-driven neural networks are right the parametric Black-Scholes formulæ overprices options for high volatility values and underprices them when the volatility is low.*

The same experiments have been repeated for shares of a number of different companies, always yielding similar results. However, there is a difference between simulating Black-Scholes options prices in the laboratory and the prices quoted on the options exchanges. The exchange-traded options on listed companies are priced using an implied volatility, which is different from the historical volatility used in the simulations.

#### 3.2 Real-Life Tests

A good way to find out whether or not neural networks are indeed correct and to establish if such patterns can be used to design a profitable statistical arbitrage trading system is by running the models through recent options data sets collected from the Chicago Board of Options Exchange.

A test set containing fourteen hundred near at the money call option prices for fifty three commonly traded American companies between August 2001 and July 2002 was used to back-test the two variants of the wavelets-based models. Only the options with between three and five days left to the expiry were selected for the test and the results are presented in Figures 6 – 12. Additional experiments have also been carried out using longer option life spans. However, the forecasting power and accuracy of the iterative models began to deteriorate rapidly when the corresponding forecasting horizon lengths were increased to over five days.

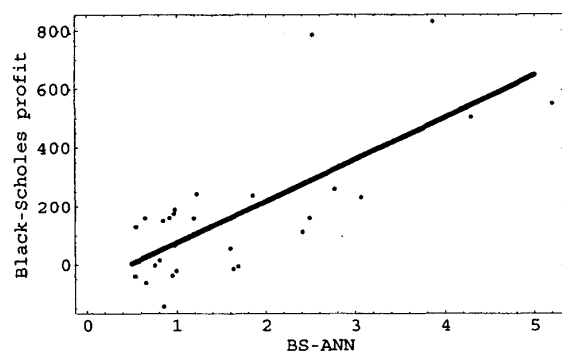


Figure 6: Profits in USD from writing hedged call options at prevailing market prices versus the difference in USD between the market price of a call option ( $BS$ ) and a neural network estimate ( $ANN$ ) for shares of 53 selected American companies (Variant A). The dots denote data points obtained from the experiment and a solid line shows a corresponding linear regression fit.

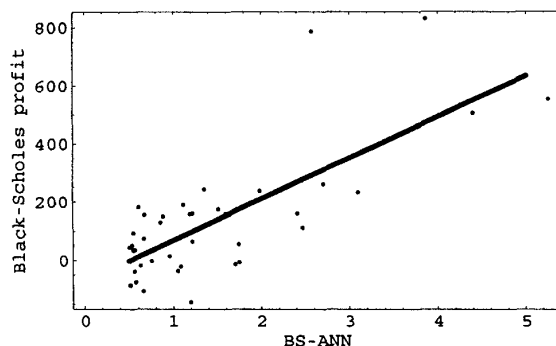


Figure 7: Profits in USD from writing hedged call options at prevailing market prices versus the difference in USD between the market price of a call option ( $BS$ ) and a neural network estimate ( $ANN$ ) for shares of 53 selected American companies (Variant B). The dots denote data points obtained from the experiment and a solid line shows a corresponding linear regression fit.

The first two Figures 6 and 7 show profits made from setting up delta-hedged trades (involving writing one call option and buying  $\Delta$  number of shares in the underlying company) versus the difference between the options prices given by the Black-Scholes and neural networks models respectively. Positions are established only when the difference between these two options prices is greater than \$0.5<sup>2</sup>. A few observations can be made after examining these two figures. First, the tangents of the regression lines in both plots are positive. This implies that the bigger the difference between the Black-Scholes estimate and a call option price provided by neural networks the bigger the profit potential on a corresponding hedged option position.

Secondly, there is a very high degree of similarity between these two charts - representing the two Variants A and B. The issue of whether or not models following the

<sup>2</sup>An explanation as to how exactly the figure of 0.5 has been arrived at will be provided further in the text

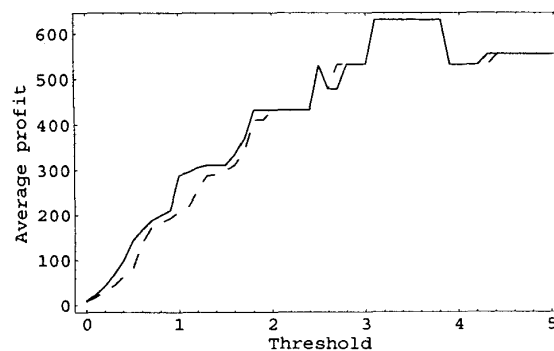


Figure 8: The average profit in USD per call option contract without transaction costs as a function of the trading threshold (solid line: Variant A and a dashed line: Variant B).

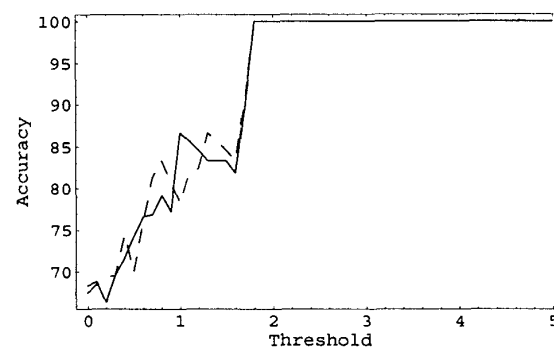


Figure 9: The percentage of correct trades as a function of the trading threshold (solid line: Variant A and a dashed line: Variant B).

Variant B can perform as well as the Variant A is important because it affects the choice of models under time performance constraints. In terms of the training and processing time, the Variant B is between five and eight times faster than the Variant A because it contains only one neural network that processes all eight wavelet components in one step. The Variant A uses eight separate networks to process the wavelets components, which could potentially enable it to produce more accurate forecasts of the wavelets coefficients.

The next five figures present various aspects of the trading performance of the two wavelet volatility models. The Figure 8 shows the average profit made from selling one delta-hedged call option contract as the trading threshold<sup>3</sup> is gradually increased. One can notice that increasing the trading threshold improves both the average profit per trade and the percentage of successful trades (see Figure 9) for both model variants. However, increasing the threshold also has the effect of reducing the total number of trades recommended by the system (as shown in Figure 10), which has

<sup>3</sup>In figures 8 – 12 the trades are only allowed when the trading threshold - as expressed by the difference between the Black-Scholes options prices  $BS$  and the corresponding neural networks estimates  $ANN$  - is greater than a pre-set value taken from the range of between zero and five

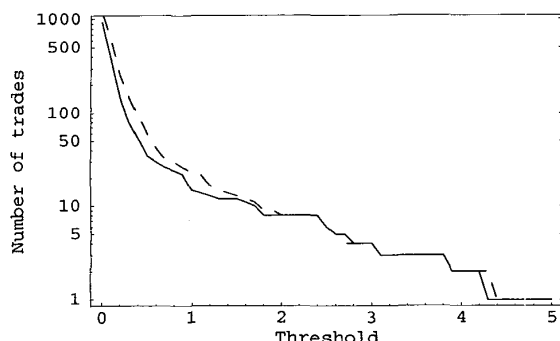


Figure 10: The total number of trades as a function of the trading threshold (solid line: Variant A and a dashed line: Variant B).

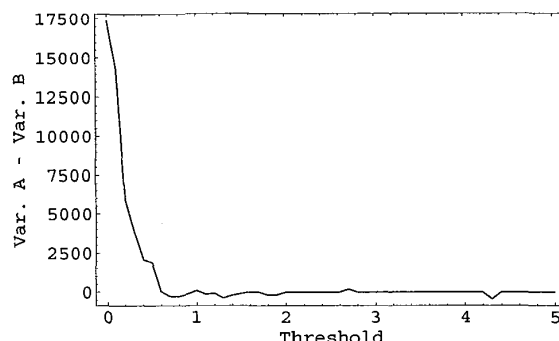


Figure 12: The difference in USD between cumulative profits for the Variants A and B as shown in Figure 11 plotted as a function of the trading threshold.

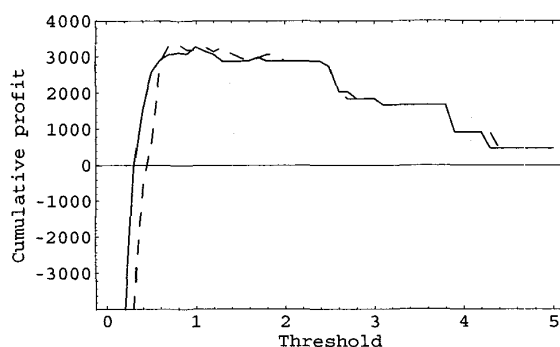


Figure 11: The cumulative profit in USD as a function of the trading threshold with fixed transaction costs of \$70 per round trade (solid line: Variant A and a dashed line: Variant B).

a detrimental effect on total cumulative profits made during the experiment. On the other hand, the presence of transaction costs introduces a further constraining factor: it makes smaller-value trades unprofitable. It would therefore be desirable to prune out loss-making small-value trades and only keep the high-value ones that can make profits over and above transaction costs.

There should be an optimum value of the threshold that would maximise the cumulative profits when transaction costs are taken into account whilst at the same time keeping the number of trades down to a minimum. A close examination of the cumulative profits as shown in Figure 11 reveals the optimum level of the threshold to be in a range of between 0.5 and 2.0. The threshold used to produce plots in Figures 6 and 7 was set to 0.5, which is at the lower end of the optimum range.

Both model variants perform similarly in trading. However, for threshold values between zero and one, the second Variant B generates more small-value trades which adversely affects the overall profitability (see Figures 11 and 12). On the other hand, the Variant B is considerably faster than the Variant A: between five and eight times.

#### 4 RESEARCH DIRECTIONS

The experimental work described in the paper could be extended and improved in a number of ways.

First, the current generation of models does not perform very well when the forecasting horizon is increased beyond five days. This is an inherent feature of any iterative forecasting model: fitting errors at each time step quickly compound which has a negative effect of degrading the forecasting accuracy the longer the forecasting horizon. Although using only short trading horizons does not prevent the models from being used profitably in performing statistical arbitrage, which is their main purpose, from a scientific point of view the evaluation would be more complete if they could be applied to pricing options over longer time horizons. Performing dynamic delta hedging is another issue that has to be considered when holding hedged options positions for longer than five days.

A potential solution to this problem could lie in changing model configurations: using a variable wavelets signal/noise cut-off point and better matching the length of the delay vectors used in the wavelets decomposition to the complexity of the signal [5]. Using longer forecasting horizons may require applying larger time scales in the wavelets signal decomposition step. For example, lengths of perhaps 64 or 128 days should be used when constructing an initial delay matrix for time horizons of fifteen days and longer. The corresponding signal/noise cut-off point (choosing how many higher-order wavelets to discard) would need to be reset accordingly. The data sampling frequency (currently set to one day) could also be an important factor in determining optimum forecasting horizons.

Secondly, the parametric Black-Scholes approach to option pricing, which has been derived under a number of restrictive assumptions, could be replaced by an alternative non-parametric model based on data-driven artificial neural networks. Initial work reported in [8] has provided a proof of concept that neural networks are quite capable of replicating the Black-Scholes formulae with a highly flexible non-linear non-parametric model.

## 5 CONCLUSIONS

The paper describes an alternative approach to modelling the stochastic volatility and its integration with conventional option pricing models.

The new volatility model uses wavelets to decompose a measure of volatility into two parts: the signal and the noise, the first one of which is then iterated forward in time with help of artificial neural networks. The volatility forecast is integrated into a traditional binomial tree in order to produce estimates for call options prices.

The models are capable of providing highly profitable opportunities for statistical arbitrage and hedging in the options markets, which may be of a potential use to the hedge fund industry.

From the profitability point of view the Variant B performs nearly as well as the Variant A. However, in terms of the processing times it has a main advantage over the Variant A: faster computing speeds and much reduced training times. This should make it a clear choice for deployment in statistical trading systems where access to good computational facilities is an issue.

Statistical arbitrage serves at least three useful purposes: it makes the markets more efficient by removing arbitrage opportunities, provides counter-parties for routine hedging trades carried out on behalf of fund managers and companies and it also provides extra liquidity needed during the periods of extreme market turbulence.

The proposed models can be employed to scan the markets on a daily basis in search of undervalued/over-valued options contracts. This would lead to more profitable statistical arbitrage trades as well as improved hedging. The techniques described in the paper could also be used to help price over the counter exotic options, for example, options that depend on more than one underlying security.

More information and sample end-of-day options prices for selected stocks can be found on the following web site: <http://www.afts-online.co.uk>.

## REFERENCES

- [1] F Black and M Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637–659, 1973.
- [2] J C Cox and S A Ross. The valuation of options for alternative stochastic processes. *Journal of Financial Economics*, 3:145–66, 1976.
- [3] M A H Dempster and A Eswaran. Wavelets methods in PDE valuation of financial derivatives. In *Proceedings of the Second International Conference of Intelligent Data Engineering and Automated Learning (IDEAL 2000)*, pages 215–238, Hong Kong, December 2000.
- [4] S Haykin. *Neural Networks*. Maxwell Macmillan, 1994.
- [5] N Hazarika and D Lowe. Iterative time-series prediction and analysis by embedding and multiple time-scale decomposition networks. In *Proceedings of SPIE, Applications and Science of Artificial Neural Networks*, pages 94–104, 1997.
- [6] J C Hull. *Options, Futures and Other Derivatives*. Prentice Hall International, 1997.
- [7] J C Hull and A White. The pricing of options on assets with stochastic volatilities. *Journal of Finance*, 42:281–300, 1987.
- [8] D Lowe. Information fusion applied to selected financial problem domains. In *Multisensor Fusion (NATO Science Series)*, volume 70, pages 749–764, 2002.
- [9] C Zapart. Stochastic volatility options pricing with wavelets and artificial neural networks. *Quantitative Finance*, 2(6):487–495, December 2002.
- [10] C Zapart and D Lowe. Non-linear iterated forecasting by neural networks with confidence intervals and sensitivity analysis. *Neural Network World*, 5:393–401, 1999.
- [11] A M Zoubir and B Boashash. The bootstrap and its application in signal processing. *IEEE Signal Processing Magazine*, 15:56–76, January 1998.